

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method for maintaining service dependency relationships between executable predefined service components ~~[[in]] of an operating system of a computer system~~ comprising the steps of:

maintaining a dynamic service consistency file in the operating system containing entries to identify predefined service components that are currently available in the operating system of the computer system, wherein:

the currently available predefined service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table; and

the linking for each currently available predefined service component entry indicates a first currently available service component from which each currently available predefined service component depends, and the linking indicates a second currently available service component that depends on the each currently available predefined service component, the each currently available predefined service component corresponding to the each currently available predefined service component entry;

comparing the dynamic service consistency file to a reference file in the operating system containing entries to identify the predefined service components ~~[[in]] of the operating system of the computer system~~, wherein:

the predefined service component entries are linked in the reference file according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table; and
the linking for each predefined service component entry in the reference file indicates a first predefined reference service component from which each predefined service component depends, and the linking indicates a second predefined reference service component that depends on the each predefined service component, the each predefined service component corresponding to the each predefined service component entry;

determining whether an inconsistency exists between service component entries within the dynamic service consistency file and the reference file; and
starting any lost predefined service component to correct any inconsistency based upon the determining step.

2. (Original) The method of claim 1 further comprising the steps of:
modifying the dynamic service consistency file based upon the starting step; and
repeating from the maintaining step.

3. (Currently Amended) The method of claim [[1]] 2 further comprising the steps of:

reading the reference file;
identifying the dependency services of the lost predefined service component according to the reference file; and
generating a log message to report the lost predefined service component including the identified dependency service components based upon the identifying the dependency services of the lost predefined service component step.

4. (Original) The method of claim 3 further comprising the step of saving the log message.

5. (Currently Amended) The method of claim [[1]] 4 further comprising the steps of:

~~reading the reference file;~~
~~identifying the dependency services of the lost predefined service component according to the reference file;~~
determining whether the lost predefined service component has been successfully started; and
generating ~~[[a]]~~ the log message to report the lost predefined service component according to the determination step, wherein the log message includes dependency service information of the lost predefined service component.

6. (Original) The method of claim 5 further comprising the steps of:
generating an alert message to report the lost predefined service according to the
determination step; and
sending the alert message to a user.

7. (Currently Amended) The method of claim [[1]] 6 wherein the maintaining step
further comprises the steps of:
identifying the currently available predefined service components; and
generating the dynamic service consistency file based upon the identifying the currently
available predefined service components step.

8. (Currently Amended) The method of claim 7 wherein the maintaining step
further comprises the steps of:
determining whether a timeout has occurred; and
repeating the identifying the currently available predefined service components step when a
timeout has occurred based upon the ~~determination~~ determining whether a timeout has
occurred step.

9. (Currently Amended) The method of claim [[1]] 8 wherein inconsistency is
based on at least one missing predefined service component in the dynamic service
consistency file when compared to the reference file.

10. (Currently Amended) The method of claim [[1]] 9 wherein the service
components include any one from the group of a process, a service hosting process, a service,
a provider service, and a dependency service.

11. (Currently Amended) The method of claim [[1]] 10 wherein the predefined
service components are installed service hosting processes or provider services that are
depended upon by other services.

12. (Canceled)

13. (Canceled)

14. (Withdrawn) A method for generating a reference file including dependency links of executable installed service components in a computer system, comprising the steps of:
reading configuration information from a registry;
identifying dependency links of the installed service components based upon the reading step;
and
generating the reference file based upon the identifying step, wherein the reference file contains entries to identify the installed service components linked according to their dependency.

15. (Withdrawn) The method according to claim 14 further comprising the steps of:
determining whether there is a change in configuration of the installed service components;
identifying the change in the installed service components based upon the determination step;
and
updating the reference file based upon the identification step.

16. (Withdrawn) A method for creating a file to indicate service dependency relationships between executable service components in a computer system, comprising:
creating a plurality of nodes to identify the service components;
wherein dependencies of the service components are linked.

17. (Withdrawn) The method of claim 16, wherein a data structure of the file is any one from the group of a tree, a graph, linked list, or a table.

18. (Withdrawn) The method of claim 16, wherein the file is a reference file containing entries to identify installed service components in the computer system or a dynamic service consistency file containing entries to identify currently available service components in the computer system.

19. (Currently Amended) A computer system for maintaining service dependency relationships between executable service components in [[a]] an operating system of the computer system, comprising:

a reference file in the operating system containing entries to identify installed service components [[in]] of the operating system of the computer system, wherein:

the installed service component entries are linked according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table; and
the linking for at least one installed service component entry indicates a first installed service component from which at least one installed service component depends, and
the linking indicates a second installed service component that depends on the at least one installed service component, the at least one installed service component
corresponding to the at least one installed service component entry;

a dynamic service consistency file in the operating system containing entries to identify currently available service components in the operating system of the computer system, wherein:

the currently available service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table; and
the linking for at least one currently available service component entry indicates a first currently available service component from which at least one currently available service component depends, and the linking indicates a second currently available service component that depends on the at least one currently available service component, the at least one currently available service component corresponding to
the at least one currently available service component entry; and

a server control manager for comparing the dynamic service consistency file to the reference file to identify any inconsistency between the files, and starting any lost service component to correct any identified inconsistency.

20. (Currently Amended) A computer-readable medium having computer-executable instructions for performing steps comprising:

maintaining a dynamic service consistency file in an operating system of a computer containing entries to identify predefined service components that are currently available in the operating system ~~a computer system~~, wherein:

the currently available predefined service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table; and

the linking for at least one currently available predefined service component entry indicates a first currently available predefined service component from which at least one currently available predefined service component depends, and the linking indicates a second currently available predefined service component that depends on the at least one currently available predefined service component, the at least one currently available predefined service component corresponding to the at least one currently available predefined service component entry;

comparing the dynamic service consistency file to a reference file in the operating system containing entries to identify the predefined service components ~~[[in]]~~ of the operating system of the computer system, wherein:

the predefined service component entries are linked in the reference file according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table; and
the linking for at least one predefined service component entry in the reference file indicates a first predefined reference service component from which at least one predefined service component depends, and the linking indicates a second predefined reference service component that depends on the at least one predefined service component, the at least one predefined service component corresponding to the at least one predefined service component entry;

determining whether an inconsistency exists between service component entries within the dynamic service consistency file and the reference file; and
starting any lost predefined service component to correct any inconsistency based upon the determining step.

21. (Withdrawn) A computer-readable medium having computer-executable instructions for performing steps comprising:
reading configuration information from a registry;
identifying dependency links of installed service components based upon the reading step;
and
generating a reference file based upon the identifying step, wherein the reference file containing entries to identify the installed service components linked according to their dependency.

22. (Withdrawn) A computer-readable medium having computer-executable instructions for performing steps comprising:
creating a plurality of nodes to identify a plurality of service components;
wherein dependencies of the service components are linked.